Rev.2.00
September 1, 2003
REJ10J0013-0200Z

Be Sure to Read This First.

SDI Emulator System for the M32R Family MCUs
M32100T-EZ-E
Release Notes

Renesas Solutions Corp.

September 1, 2003

Outline

These release note explain the specifications, cautions and restrictions dependent on MCU models which are not covered by the M32100T-EZ-E User's Manual. When referring to a required section in the user's manual, read the release notes together with the user's manual.

Contents

1. MCU Model List 1
2. Specifications, Cautions and Restrictions Dependent on the MCU Model 1
2.1 32102 Group 2
2.2 32104 Group 4
2.3 32121 Group 6
2.4 2170, 32171, 32172 ,32173 and 32174 Groups 9
2.5 32180 and 32182 Groups 10

1. MCU Model List

These release notes cover the MCU models listed in Table 1.

Table 1 MCU model list

32102 Group	M32102S6FP
32104 Group	M32104S6FP
32121 Group	M32121FCWG
32170 Group	M32170FxVFP
32171 Group	M32171FxVFP
32172 Group	M32172FxVFP, M32172FxVWG
32173 Group	M32173FxVFP, M32173FxVWG
32174 Group	M32174FxVFP
32180 Group	M32180FxxFP
32182 Group	M32182FxxFP

2. Specifications, Cautions and Restrictions Dependent on the MCU Model

The following explains the emulator specifications, cautions and restrictions dependent on the MCU model.

2.1 32102 Group

(1) Differences between the emulator and MCUs

IMPORTANT

- The emulator initializes the MCU internal register to the following values. Initialization starts when the power switch is turned ON or MCU is reset.

R0 = 00000000 R1 = 00000000 R2 = 00000000 R3 = 00000000 R4 = 00000000
R5 = 00000000 R6 = 00000000 R7 = 00000000 R8 = 00000000 R9 = 00000000
R10 = 00000000 R11 = 00000000 R12 = 00000000 R13 = 00000000 R14 = 00000000
R15 = 00000100 SPI = 00000100 SPU = 00000100 BPC = 00000000 PC = 00000000
ACCH = xxxxxxxx ACCL = xxxxxxxx
PSW = 00000000

(2) MCU signals connected to the SDI MCU control interface connector (Section 3.2)

Pin No.	Pin	Direction	Connected to	Remarks
1	TCLK	Emulator to target	TCK of MCU	Clock frequency: 5 MHz
2	Vss	-	GND (0 V)	
3	TDI	Emulator to target	TDI of MCU	
4	TDO	Target to emulator	TDO of MCU	
5	TMS	Emulator to target	TMS of MCU	
6	TRST	Emulator to target	TRST# of MCU	Totem pole output on the emulator
7	DBI	Emulator to target	DBI# of MCU	
8	N.C.	-	Not used	
9	Vcc	Target to emulator	VCCX of MCU	
10	RST	Emulator to target	System reset	Open corrector output on the emulator

(3) Debug specifications dependent on MCU model (Section 5.1)

Emulation memory		• No built-in emulation memory provided in the emulator • Function to download the data to an external flash ROM supported
Software break		RAM area: Implemented by instruction replacement ROM area: Implemented by preexecution PC breakpoints of MCU (4 points)
Hardware break	Forcible break	Implemented by MCU's internal resources
	Data access break	Implemented by MCU's internal resources (2 data access break points)

(4) Other cautions and restrictions dependent on MCU model

IMPORTANT

Note on Using the MVTC Instruction (1):

- When the user program is stopped by the MVTC instruction at the address of the instruction which operates BPC, a BPC value may be changed even if the MVTC instruction has not been executed. Occurrence of this phenomenon depends on the MCU status and the instruction sequence. After this phenomenon occurs, the program is normally re-executed starting with the PC value of the MVTC instruction.
  The following shows sample data when the phenomenon occurs.

  Example:

```
Address          Instruction
H'100    LD24    R0,#H'100
H'104    MVTC    R0,BPC
H'106    NOP
  .
  .
```

  When a break occurs at PC = H'104 in the above program fragment, BPC = H'100 may be enabled even before the MVTC instruction is executed.

Note on Using the MVTC Instruction (2):

- In cases when instructions used to operate on the BPC with the MVTC instruction are followed by an RTE instruction, do not set a preexecution PC breakpoint for the RTE instruction. This is because the BPC value becomes illegal and the program cannot be run normally after a break.
  The following shows sample data when the phenomenon occurs.

  Example:

```
Address          Instruction
H'100    LD24    R0,#H'100
H'104    MVTC    R0,BPC
H'106    NOP
  .
  .
H'110    RTE
```

  If in the above program fragment a preexecution PC breakpoint is set at PC = H'110 to cause a break, the BPC value becomes illegal.

Note on WDT:

- If a user program is stopped during operation of WDT built in the MCU, the WDT count halts meanwhile. This prevents SBI interruption by WDT while the user program is stopped. However, WDT is normally initialized then.

2.2 32104 Group

(1) Differences between the emulator and MCUs

IMPORTANT

- The emulator initializes the MCU internal register to the following values. Initialization starts when the power switch is turned ON or MCU is reset.

R0 = 00000000 R1 = 00000000 R2 = 00000000 R3 = 00000000 R4 = 00000000
R5 = 00000000 R6 = 00000000 R7 = 00000000 R8 = 00000000 R9 = 00000000
R10 = 00000000 R11 = 00000000 R12 = 00000000 R13 = 00000000 R14 = 00000000
R15 = 00000100 SPI = 00000100 SPU = 00000100 BPC = 00000000 PC = 00000000
ACCH = xxxxxxxx ACCL = xxxxxxxx
PSW = 00000000

(2) MCU signals connected to the SDI MCU control interface connector (Section 3.2)

Pin No.	Pin	Direction	Connected to	Remarks
1	TCLK	Emulator to target	TCK of MCU	Clock frequency: 5 MHz
2	Vss	-	GND (0 V)	
3	TDI	Emulator to target	TDI of MCU	
4	TDO	Target to emulator	TDO of MCU	
5	TMS	Emulator to target	TMS of MCU	
6	TRST	Emulator to target	TRST# of MCU	Totem pole output on the emulator
7	DBI	Emulator to target	DBI# of MCU	Not used when using M32104S6FP
8	N.C.	-	Not used	
9	Vcc	Target to emulator	VCCX of MCU	
10	RST	Emulator to target	System reset	Open corrector output on the emulator

(3) Debug specifications dependent on MCU model (Section 5.1)

Emulation memory		• No built-in emulation memory provided in the emulator • Function to download the data to an external flash ROM supported
Software break		RAM area: Implemented by instruction replacement ROM area: Implemented by preexecution PC breakpoints of MCU (4 points)
Hardware break	Forcible break	Implemented by MCU's internal resources
	Data access break	Implemented by MCU's internal resources (2 data access break points)

(4) Other cautions and restrictions dependent on MCU model

IMPORTANT

Note on Using the MVTC Instruction (1):

- When the user program is stopped by the MVTC instruction at the address of the instruction which operates BPC, a BPC value may be changed even if the MVTC instruction has not been executed. Occurrence of this phenomenon depends on the MCU status and the instruction sequence. After this phenomenon occurs, the program is normally re-executed starting with the PC value of the MVTC instruction.
  The following shows sample data when the phenomenon occurs.

  Example:

```
Address          Instruction
H'100    LD24    R0,#H'100
H'104    MVTC    R0,BPC
H'106    NOP
  .
  .
```

  When a break occurs at PC = H'104 in the above program fragment, BPC = H'100 may be enabled even before the MVTC instruction is executed.

Note on Using the MVTC Instruction (2):

- In cases when instructions used to operate on the BPC with the MVTC instruction are followed by an RTE instruction, do not set a preexecution PC breakpoint for the RTE instruction. This is because the BPC value becomes illegal and the program cannot be run normally after a break.
  The following shows sample data when the phenomenon occurs.

  Example:

```
Address          Instruction
H'100    LD24    R0,#H'100
H'104    MVTC    R0,BPC
H'106    NOP
  .
  .
H'110    RTE
```

  If in the above program fragment a preexecution PC breakpoint is set at PC = H'110 to cause a break, the BPC value becomes illegal.

2.3 32121 Group

(1) Differences between the emulator and MCUs

IMPORTANT

- The emulator initializes the MCU internal register to the following values. Initialization starts when the power switch is turned ON or MCU is reset.

  R0 = 00000000 R1 = 00000000 R2 = 00000000 R3 = 00000000 R4 = 00000000
  R5 = 00000000 R6 = 00000000 R7 = 00000000 R8 = 00000000 R9 = 00000000
  R10 = 00000000 R11 = 00000000 R12 = 00000000 R13 = 00000000 R14 = 00000000
  R15 = 00000100 SPI = 00000100 SPU = 00000100 BPC = 00000000 PC = 00000000
  ACCH = xxxxxxxx ACCL = xxxxxxxx
  PSW = 00000000

- To execute a target program which transfers to the sleep mode or to the stop mode, you must observe the following restrictions because of the MCU specifications:

  (1) Add the instructions given below after the store instruction which transfers the target program to the sleep/stop mode.
  (2) Do not set a break point between the STB instruction and the last NOP instruction added.

```
    Target program
                .
                .
                Store instruction   <--Instruction to transfer to the sleep/stop mode
LOOP:           BRA   LOOP          \
                NOP                 |
                NOP                 |
                .                   |  Insert 16 NOP instructions after the address branch command.
                .                   |
                NOP                 /
```

- When a break is forced while the target MCU is set in the sleep/stop mode, the target MCU may return to the normal mode. Also when memory is referenced or set while the target MCU is set in the sleep/stop mode, the target MCU may return to the normal mode from the sleep/stop mode.
- Do not change the target MCU to the sleep/stop mode using the memory rewrite function of the emulator debugger.

(2) MCU signals connected to the SDI MCU control interface connector (Section 3.2)

Pin No.	Pin	Direction	Connected to	Remarks
1	TCLK	Emulator to target	TCK of MCU	Clock frequency: 5 MHz
2	Vss	-	GND (0 V)	
3	TDI	Emulator to target	TDI of MCU	
4	TDO	Target to emulator	TDO of MCU	
5	TMS	Emulator to target	TMS of MCU	
6	TRST	Emulator to target	TRST# of MCU	Totem pole output on the emulator
7	DBI	Emulator to target	DBI# of MCU	
8	FVCC	Emulator to target	FVCC of MCU	Voltage is set to 2.5 V. Turned ON/OFF according to the setting of the FVCC switch.
9	Vcc	Target to emulator	VCCJ of MCU	
10	RST	Emulator to target	System reset	Open corrector output on the emulator

(3) Debug specifications dependent on MCU model (Section 5.1)

Emulation memory		Flash ROM built in MCU used as emulation memory
Software break		RAM area: Implemented by instruction replacement ROM area: Implemented by preexecution PC breakpoints of MCU (4 points)
Hardware break	Forcible break	Implemented by MCU's internal resources
	Data access break	Implemented by MCU's internal resources (2 data access break points)

(4) Other cautions and restrictions dependent on MCU model

IMPORTANT

Note on Using the MVTC Instruction (1):

- When the user program is stopped by the MVTC instruction at the address of the instruction which operates BPC, a BPC value may be changed even if the MVTC instruction has not been executed. Occurrence of this phenomenon depends on the MCU status and the instruction sequence. After this phenomenon occurs, the program is normally re-executed starting with the PC value of the MVTC instruction.
  The following shows sample data when the phenomenon occurs.

  Example:

```
Address              Instruction
H'100     LD24       R0,#H'100
H'104     MVTC       R0,BPC
H'106     NOP
            .
            .
            .
```

  When a break occurs at PC = H'104 in the above program fragment, BPC = H'100 may be enabled even before the MVTC instruction is executed.

Note on Using the MVTC Instruction (2):

- In cases when instructions used to operate on the BPC with the MVTC instruction are followed by an RTE instruction, do not set a preexecution PC breakpoint for the RTE instruction. This is because the BPC value becomes illegal and the program cannot be run normally after a break.
  The following shows sample data when the phenomenon occurs.

  Example:

```
Address              Instruction
H'100     LD24       R0,#H'100
H'104     MVTC       R0,BPC
H'106     NOP
            .
            .
            .
H'110     RTE
```

  If in the above program fragment a preexecution PC breakpoint is set at PC = H'110 to cause a break, the BPC value becomes illegal.

Note on WDT:

- If a user program is stopped during operation of WDT built in the MCU, the WDT count halts meanwhile. This prevents SBI interruption by WDT while the user program is stopped. However, WDT is normally initialized then.

2.4 32170, 32171, 32172, 32173 and 32174 Groups

(1) Differences between the emulator and MCUs

IMPORTANT

- The emulator initializes the MCU internal register to the following values. Initialization starts when the power switch is turned ON or MCU is reset.

  R0 = 00000000 R1 = 00000000 R2 = 00000000 R3 = 00000000 R4 = 00000000
  R5 = 00000000 R6 = 00000000 R7 = 00000000 R8 = 00000000 R9 = 00000000
  R10 = 00000000 R11 = 00000000 R12 = 00000000 R13 = 00000000 R14 = 00000000
  R15 = 00000100 SPI = 00000100 SPU = 00000100 BPC = 00000000 PC = 00000000
  ACCH = xxxxxxxx ACCL = xxxxxxxx
  PSW = 00000000

(2) MCU signals connected to the SDI MCU control interface connector (Section 3.2)

Pin No.	Pin	Direction	Connected to	Remarks
1	TCLK	Emulator to target	JTCK of MCU	Clock frequency: 4 MHz
2	Vss	-	GND (0 V)	
3	TDI	Emulator to target	JTDI of MCU	
4	TDO	Target to emulator	JTDO of MCU	
5	TMS	Emulator to target	JTMS of MCU	
6	TRST	Emulator to target	JTRST of MCU	Totem pole output on the emulator
7	DBI	Emulator to target	JDBI of MCU	Not used when using M3217xFxxFP
8	N.C.	-	Not used	
9	Vcc	Target to emulator	VCCE of MCU	
10	RST	Emulator to target	System reset	Open corrector output on the emulator

(3) Debug specifications dependent on MCU model (Section 5.1)

Emulation memory		Flash ROM built in MCU used as emulation memory
Software break		RAM area: Implemented by instruction replacement ROM area: Implemented by preexecution PC breakpoints of MCU (4 points)
Hardware break	Forcible break	Implemented by MCU's internal resources
	Data access break	Implemented by MCU's internal resources (2 data access break points)

(4) Other cautions and restrictions dependent on MCU model

IMPORTANT

Note on Access Prohibited Areas:

- The emulator uses the system space from H'C000 0000 to H'FFFF FFFF stipulated in the MCU user's manual as an emulator-only area. If the system space is accessed by the target program, operation of the emulator cannot be guaranteed. Therefore, do not access the area H'C000 0000 through H'FFFF FFFF from the target program.

2.5 32180 and 32182 Groups

(1) Differences between the emulator and MCUs

IMPORTANT

- The emulator initializes the MCU internal register to the following values. Initialization starts when the power switch is turned ON or MCU is reset.

  R0 = 00000000 R1 = 00000000 R2 = 00000000 R3 = 00000000 R4 = 00000000
  R5 = 00000000 R6 = 00000000 R7 = 00000000 R8 = 00000000 R9 = 00000000
  R10 = 00000000 R11 = 00000000 R12 = 00000000 R13 = 00000000 R14 = 00000000
  R15 = 00000100 SPI = 00000100 SPU = 00000100 BPC = 00000000 PC = 00000000
  ACCH = xxxxxxxx ACCL = xxxxxxxx
  PSW = 00000000

(2) MCU signals connected to the SDI MCU control interface connector (Section 3.2)

Pin No.	Pin	Direction	Connected to	Remarks
1	TCLK	Emulator to target	JTCK of MCU	Clock frequency: 5 MHz
2	Vss	-	GND (0 V)	
3	TDI	Emulator to target	JTDI of MCU	
4	TDO	Target to emulator	JTDO of MCU	
5	TMS	Emulator to target	JTMS of MCU	
6	TRST	Emulator to target	JTRST of MCU	Totem pole output on the emulator
7	DBI	Emulator to target	JDBI of MCU	Not used when using M3218xFxxFP
8	FVCC	Emulator to target	SDIVCC of MCU	Not used when using M3218xFxxFP
9	Vcc	Target to emulator	VCCE of MCU	
10	RST	Emulator to target	System reset	Open corrector output on the emulator

(3) Debug specifications dependent on MCU model (Section 5.1)

Emulation memory		Flash ROM built in MCU used as emulation memory
Software break		RAM area: Implemented by instruction replacement ROM area: Implemented by preexecution PC breakpoints of MCU (4 points)
Hardware break	Forcible break	Implemented by MCU's internal resources
	Data access break	Implemented by MCU's internal resources (2 data access break points)

(4) Other cautions and restrictions dependent on MCU model

IMPORTANT

Note on Access Prohibited Areas:

- The emulator uses the system space from H'C000 0000 to H'FFFF FFFF stipulated in the MCU user's manual as an emulator-only area. If the system space is accessed by the target program, operation of the emulator cannot be guaranteed. Therefore, do not access the area H'C000 0000 through H'FFFF FFFF from the target program.